In the claims:

1. A node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

generating a new spanning tree after a network configuration change while continuing to operate the spanning tree that existed before the configuration change, and switching the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

2. The node as set forth in claim 1 wherein

said network configuration change is addition or remove of a node or a change in link topology.

3. A node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

generating, at the time of a link cost change of the network, a new spanning tree after the cost change while continuing to operate an existing spanning tree, and switching the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

4. A node that configures a spanning tree over a

network to which a plurality of nodes are connected, comprising:

a plurality of tree managers that generate a plurality of independently operating spanning trees,

a tag table that returns a tag corresponding to the spanning tree that is used for forwarding,

a tag insertion unit that inserts the tag that has been returned from said tag table into a frame,

a tree selector that determines the spanning tree used for forwarding,

a forwarding table in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding unit that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator that determines the tree manager of the forwarding destination of said frame according to said tag.

5. The node as set forth in claim 4 wherein

said tree selector comprises:

a main controller that performs switching of the spanning tree used for forwarding,

a stable timer that notifies of the expiration of the timer for a specified time, which indicates stabilization of the spanning tree,

a tag remove unit that removes the tag that has

been added to the frame,

a GVRP transmitter/receiver that transmits a control frame to switch spanning trees, and

a tag insertion unit that adds a tag to the frame.

6. The node as set forth in claim 5 wherein said tree selector comprises

an arrival interval timer that sends a timer expiration notice after a given length of time has elapsed, in order to determine the frame arrival intervals, which indicate stabilization of the spanning tree.

7. The node as set forth in claim 4 wherein said tree selector comprises

a cost reference timer that notifies of the expiration of the timer for a specified time used for the calculation of link cost.

8. The node as set forth in claim 4 wherein said tree manager comprises:

a tag remove unit that removes the tag that has been added to the frame,

a BPDU transmitter/receiver that transmits and receives a BPDU,

a tag insertion unit that adds a tag to the frame,

a tree controller that creates the spanning tree

according to a spanning tree protocol, and

a tree table that retains parameters used in said spanning tree protocol.

9. The node as set forth in claim 8 wherein said tree manager comprises a cost operator that adds a prescribed setting value to the link cost that has been notified and returns it.

10. The node as set forth in claim 4 comprising a resource monitor that measures resource information including the connection status and the free bandwidth of a link.

11. The node as set forth in claim 3 wherein said link cost is calculated based on the availability status.

12. The node as set forth in claim 11 wherein said availability status is defined as a free bandwidth.

13. The node as set forth in claim 11 wherein said availability status is defined as a CPU load.

14. A node that configures a spanning tree over a

network to which a plurality of nodes are connected, comprising

generating a spanning tree in which each node in the network serves as a root node, and forwarding a frame (frames) using a spanning tree in which the destination serves as a root node.

15. A node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

a plurality of tree managers that generate a plurality of independently operating spanning trees,

a tag table that returns a tag corresponding to the spanning tree that is used for forwarding,

a tag insertion unit that inserts the tag that has been returned from said tag table into a frame,

a tree selector that generates as many tree managers as the number of root nodes that exist in the network,

a forwarding table in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding unit that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator that determines the tree manager of the forwarding destination of said frame according to said tag.

16. The node as set forth in claim 15 wherein said tree selector comprises:

a main controller that creates or removes the tree manager,

a tag remove unit that removes the tag that has been added to the frame,

a GVRP transmitter/receiver that transmits a control frame to switch spanning trees, and

a tag insertion unit that adds a tag to the frame.

17. The node as set forth in claim 15 wherein said tree manager comprises:

a tag remove unit that removes the tag that has been added to the frame,

a BPDU transmitter/receiver that transmits and receives a BPDU,

a tag insertion unit that adds a tag to the frame,

a tree controller that creates the spanning tree according to a spanning tree protocol, and

a tree table that retains parameters used in said spanning tree protocol.

18. The node as set forth in claim 15 comprising a resource monitor that measures resource information including the connection status and the free bandwidth of a link.

19. A node that configures a spanning tree over a network to which a plurality of nodes are connected wherein

a tree manager that generates the spanning tree comprises

a cost operator that adjusts a cost value based on the type and the version of a spanning tree protocol.

20. The node as set forth in claim 19 wherein

said cost operator

allocates a high cost to a link that uses a protocol whose failure recovery processing is slow.

21. A node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising

generating a spanning tree in which the cost of each link is maximum for each link that exists in the network and that uses a protocol whose operation is slow and in case a failure occurs at said each link,

forwarding a frame using the tree in which the cost of said link is maximum.

22. A node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

a plurality of tree managers that generate a plurality of independently operating spanning trees,

a tag table that returns a tag corresponding to the tree that is used for forwarding,

a tag insertion unit that inserts the tag that has been returned from said tag table into a frame,

a tree selector that generates as many tree managers as the number of links that exist in the network and use a protocol whose operation is slow,

a forwarding table in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding unit that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator that determines the tree manager of the forwarding destination according to said tag.

23. The node as set forth in claim 22 wherein said tree selector comprises:

a main controller in the tree selector that creates or removes the tree manager,

a tag remove unit that removes the tag that has been added to the frame,

a GVRP transmitter/receiver that transmits a control frame, and

a tag insertion unit that adds a tag to the frame.

24. The node as set forth in claim 22 wherein said tree manager comprises:

a tag remove unit that removes the tag that has been added to the frame,

a BPDU transmitter/receiver that transmits and receives a BPDU,

a tag insertion unit that adds a tag to the frame,

a tree controller that creates the spanning tree according to a spanning tree protocol, and

a tree table that retains parameters used in the spanning tree protocol.

25. The node as set forth in claim 22 comprising a resource monitor that measures resource information including the connection status and the free bandwidth of a link.

26. The node as set forth in claim 4 comprising a failure detector that transmits and receives frames for failure detection at intervals shorter than those of HELLO frames that are used by the spanning tree protocol to detect a failure.

27. The node as set forth in claim 4 wherein said forwarding table possesses a broadcast output port field.

28. The node as s t forth in claim 4 wherein

said forwarding table possesses an auxiliary output port field.

29. The node as set forth in claim 4 wherein

an output destination port is determined using a port type determined by the spanning tree.

30. The node as set forth in claim 29 wherein

the port type determined by said spanning tree is either one of a Root Port or a Designated Port.

31. A spanning tree configuration program that operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising

a function that generates a new spanning tree after a network configuration change while continuing to operate the spanning tree that existed before the configuration change, and switches the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

32. The spanning tree configuration program as set forth in claim 31 wherein

said network configuration change is addition or of a node or a change in link topology.

33. A spanning tree configuration program that operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising

a function that generates, at the time of a link cost change of the network, a new spanning tree after the cost change while continuing to operate an existing spanning tree, and switches the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

34. A spanning tree configuration program that operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

a function that generates a plurality of independently operating spanning trees, via a plurality of tree managers,

a function that returns a tag corresponding to the spanning tree that is used for forwarding,

a tag insertion function that inserts said tag that has been returned into a frame,

a tree selector function that determines the tree used for forwarding,

a forwarding table function in which a forwarding output destination of the frame is recorded by

destination,

a frame forwarding function that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator function that determines the tree manager of the forwarding destination according to said tag.

35. The spanning tree configuration program as set forth in claim 34 wherein

said tree selector function executes:

a controller function that performs switching of the spanning tree used for forwarding,

a stable timer function that notifies of the expiration of the timer for a specified time, which indicates stabilization of the spanning tree,

a tag remove function that removes the tag that has been added to the frame,

a GVRP transmitter/receiver function that transmits a control frame to switch spanning trees, and

a tag insertion function that adds a tag to the frame.

36. The spanning tree configuration program as set forth in claim 35 wherein

said tree selector function executes

an arrival interval timer function that sends a

timer expiration notice after a given length of time has elapsed, in order to determine the frame arrival intervals, which indicate stabilization of the spanning tree.

37. The spanning tree configuration program as set forth in claim 34 wherein

said tree selector function executes:

a cost reference timer function that notifies of the expiration of the timer for a specified time used for the calculation of link cost.

38. The spanning tree configuration program as set forth in claim 34 wherein

said tree manager function executes:

a tag remove function that removes the tag that has been added to the frame,

a BPDU transmitter/receiver function that transmits and receives a BPDU,

a tag insertion function that adds a tag to the frame,

a tree controller function that creates the spanning tree according to a spanning tree protocol, and

a tree table function that retains parameters used in said spanning tree protocol.

39. The spanning tree configuration program as set

forth in claim 38 wherein

said tree manager function executes

a cost operator function that adds a prescribed setting value to the link cost that has been notified and returns it.

40. The spanning tree configuration program as set forth in claim 34

executing a resource monitor function that measures resource information including the connection status and the free bandwidth of a link.

41. The spanning tree configuration program as set forth in claim 33

executing a function that calculates the link cost based on the availability status.

42. The spanning tree configuration program as set forth in claim 41 wherein

said availability status is defined as a free bandwidth.

43. The spanning tree configuration program as set forth in claim 41 wherein

said availability status is defined as a CPU load.

44. A spanning tree configuration program that

operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising

a function that generates a spanning tree in which each node in the network serves as a root node, and forwards a frame using a tree in which the destination serves as a root node.

45. A spanning tree configuration program that operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

a plurality of tree manager functions that generate a plurality of independently operating spanning trees,

a tag table function that returns a tag corresponding to the spanning tree that is used for forwarding,

a tag insertion function that inserts the tag that has been returned from said tag table into a frame,

a tree selector function that generates as many tree managers as the number of root nodes that exist in the network,

a forwarding table function in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding function that forwards the

frame to the forwarding output destination that is specified in said forwarding table, and

a separator function that determines the tree manager of the forwarding destination of said frame according to said tag.

46. The spanning tree configuration program as set forth in claim 45 wherein

said tree selector function executes:

a main controller function in the tree selector that creates or removes the tree manager,

a tag remove function that removes the tag that has been added to the frame,

a GVRP transmitter/receiver function that transmits a control frame to switch spanning trees, and

a tag insertion function that adds a tag to the frame.

47. The spanning tree configuration program as set forth in claim 45 wherein

said tree manager function executes:

a tag remove function that removes the tag that has been added to the frame,

a BPDU transmitter/receiver function that transmits and receives a BPDU,

a tag insertion function that adds a tag to the frame,

a tree controller function that creates the spanning tree according to a spanning tree protocol, and

a tree table function that retains parameters used in said spanning tree protocol.

48. The spanning tree configuration program as set forth in claim 45 wherein

each of said nodes executes a resource monitor function that measures resource information including the connection status and the free bandwidth of a link.

49. A spanning tree configuration program that operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected

generating a spanning tree in which the cost of each link is maximum for each link that exists in the network and that uses a protocol whose operation is slow and in case a failure occurs at said each link,

forwarding a frame using the tree in which the cost of said link is maximum.

50. A spanning tree configuration program that operates on each node that configures a spanning tree over a network to which a plurality of nodes are connected, comprising:

a plurality of tree manager functions that

generate a plurality of independently operating spanning trees,

a tag table function that returns a tag corresponding to the tree that is used for forwarding,

a tag insertion function that inserts the tag that has been returned from said tag table into a frame,

a tree selector function that generates as many tree managers as the number of links that exist in the network and use a protocol whose operation is slow,

a forwarding table function in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding function that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator function that determines the tree manager of the forwarding destination of the frame according to said tag.

51. The spanning tree configuration program as set forth in claim 50 wherein

said tree selector function comprises:

a main controller function in the tree selector that creates or removes the tree manager,

a tag remove function that removes the tag that has been added to the frame,

a GVRP transmitter/receiver function that

transmits a control frame, and

a tag insertion function that adds a tag to the frame.

52. The spanning tree configuration program as set forth in claim 50 wherein

said tree manager function comprises:

a tag remove function that removes the tag that has been added to the frame,

a BPDU transmitter/receiver function that transmits and receives a BPDU,

a tag insertion function that adds a tag to the frame,

a tree controller function that creates the spanning tree according to a spanning tree protocol, and

a tree table function that retains parameters used in the spanning tree protocol.

53. The spanning tree configuration program as set forth in claim 50 wherein

each of said nodes executes a resource monitor function that measures resource information including the connection status and the free bandwidth of a link.

54. The spanning tree configuration program as set forth in claim 34 wherein

said forwarding table possesses a broadcast

output port field.

55. The spanning tree configuration program as set forth in claim 34 wherein

said forwarding table possesses an auxiliary output port field.

56. The spanning tree configuration program as set forth in claim 34 wherein

an output destination port is determined using a port type determined by the spanning tree.

57. The spanning tree configuration program as set forth in claim 56 wherein

the port type determined by said spanning tree is either one of a Root Port or a Designated Port.

58. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected wherein

each of said nodes

generates a new spanning tree after a network configuration change while continuing to operate the spanning tree that existed before the configuration change, and switches the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

59. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected wherein

each of said nodes

generates, at the time of a link cost change of the network, a new spanning tree after the cost change while continuing to operate an existing spanning tree, and switches the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

60. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected wherein

each of said nodes comprises:

a plurality of tree managers that generate a plurality of independently operating spanning trees,

a tag table that returns a tag corresponding to the spanning tree that is used for forwarding,

a tag insertion unit that inserts the tag that has been returned from said tag table into a frame,

a tree selector that determines the spanning tree used for forwarding,

a forwarding table in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding unit that forwards the frame

to the forwarding output destination that is specified in said forwarding table, and

a separator that determines the tree manager of the forwarding destination of said frame according to said tag.

61. The network system as set forth in claim 59 wherein

the link cost is calculated based on the availability status.

62. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected, comprising:

generating a spanning tree in which each node in the network serves as a root node, and forwarding a frame using a tree in which the destination serves as a root node.

63. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected, comprising:

a plurality of tree managers that generate a plurality of independently operating spanning trees,

a tag table that returns a tag corresponding to the tree that is used for forwarding,

a tag insertion unit that inserts the tag that

has been returned from said tag table into a frame,

a tree selector that generates as many tree managers as the number of nodes that exist in the network,

a forwarding table in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding unit that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator that determines the tree manager of the forwarding destination of said frame according to said tag.

64. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected wherein

a tree manager that generates the spanning tree executes

a cost operation processing that adjusts a cost value based on the type and the version of a spanning tree protocol.

65. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected wherein

a tree manager that generates the spanning tree comprises

a cost operator that adjusts a cost value based on the type and the version of a spanning tree protocol.

66. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected, comprising:

generating a spanning tree in which the cost of each link is maximum for each link that exists in the network and that uses a protocol whose operation is slow and in case a failure occurs at said each link, forwarding a frame using the tree in which the cost of said link is maximum.

67. A network system in which a forwarding path is set by a spanning tree over a network to which a plurality of nodes are connected, comprising:

a plurality of tree managers that generate a plurality of independently operating spanning trees,

a tag table that returns a tag corresponding to the tree that is used for forwarding,

a tag insertion unit that inserts the tag that has been returned from said tag table into a frame,

a tree selector that generates as many tree managers as the number of links that exist in the network and use a protocol whose operation is slow,

a forwarding table in which a forwarding output destination of the frame is recorded by destination,

a frame forwarding unit that forwards the frame to the forwarding output destination that is specified in said forwarding table, and

a separator that determines the tree manager of the forwarding destination of said frame according to said tag.

68. The network system as set forth in claim 60 wherein

said forwarding table possesses a broadcast output port field.

69. The network system as set forth in claim 60 wherein

said forwarding table possesses an auxiliary output port field.

70. The network system as set forth in claim 60 wherein

an output destination port is determined using a port type determined by the spanning tree.

71. The network system as set forth in claim 70 wherein

the port type determined by said spanning tree is either one of a Root Port or a Designated Port.

72. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the steps of:

generating a new spanning tree after a network configuration change while continuing to operate the spanning tree that existed before the configuration change, and switching the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

73. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the steps of:

generating, at the time of a link cost change of the network, a new spanning tree after the cost change while continuing to operate an existing spanning tree, and switching the spanning tree to be used for forwarding to said new spanning tree after said new spanning tree has been stable.

74. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the step of:

making a new node participate in an auxiliary spanning tree only, not in an existing spanning tree, when adding the new node.

75. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the step of:

making a removing node participate in an existing spanning tree only, not in an auxiliary spanning tree, when removing the node.

76. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the step of:

creating a tree after a change using an auxiliary system, when a network configuration has changed.

77. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the step of:

using a link free bandwidth to calculate a cost.

78. A spanning tree configuration method in a network to which a plurality of nodes are connected, comprising the step of:

creating a plurality of spanning trees so that all the nodes in the network serve as the root node of any one spanning tree among the spanning trees that have all the nodes as members.

79. A spanning tree configuration method in a network

to which a plurality of nodes are connected, comprising the steps of:

creating spanning trees that have all the nodes that exist in the network as members, and, among them, creating a plurality of spanning trees for each link that uses a protocol whose failure recovery is slow.

80. Method of forming a logical topology that is used for signal transmission in a network to which a plurality of nodes are connected, comprising the steps of:

generating a logical topology after a network configuration change with the signal transmission being performed using the logical topology that existed before the network configuration change, and

after the logical topology after said configuration change has been stable, switching the logical topology to be used for signal transmission to the logical topology after said configuration change.

81. A node comprising

a element which generates a logical topology after a network configuration change, when changing the configuration of a network to which it belongs itself, with the signal transmission being performed using the logical topology in said network, and

a element which switches, after the logical

topology after said configuration change has been stable, the logical topology to be used for signal transmission to the logical topology after said configuration change.

82. A program comprising a function of generating a logical topology after a network configuration change, when changing the configuration of a network to which it belongs itself, with the signal transmission being performed using the logical topology in said network, and

a function of switching, after the logical topology after said configuration change has been stable, the logical topology to be used for signal transmission to the logical topology after said configuration change.

83. A network system to which a plurality of nodes are connected, comprising:

generating a logical topology after a network configuration change with the signal transmission being performed using the logical topology that existed before the network configuration change, and

after the logical topology after said configuration change has been stable, switching the logical topology to be used for signal transmission to the logical topology after said configuration change.

84. A node comprising a element which generates a

correspondence between the information on a destination, which a frame to be entered retains, and a forwarding destination of said frame using a spanning tree protocol, and

a element which refers to said correspondence to determine the forwarding destination of the frame that has been entered.